Claims

The following is a copy of Applicant's claims that identifies language being added with underlining ("___") and language being deleted with strikethrough ("—"), as is applicable:

1-6. (Canceled)

7. (Currently amended) A method ~~for processing breakpoint events in a child process created from a parent process, wherein the parent process is modified by a software tool, the method~~ comprising:

instrumenting a parent process to generate instrumented functions in the parent process so as to create an instrumented version of the parent process;

storing uninstrumented instructions associated with the instrumented functions of the parent process;

~~storing unmodified uninstrumented parent process code replaced by each occurrence of a breakpoint inserted into the address space during modification of the parent process;~~

monitoring with a process monitor execution of a child process created by the parent process for an initial breakpoint in the address space of the child process;

suspending with the process monitor execution of the child process in response to detection of the initial breakpoint with the process monitor;

replacing with the process monitor each occurrence of a breakpoint in the address space of the child process with the ~~unmodified~~ uninstrumented ~~parent process code~~ instructions such that the child process reflects an original, uninstrumented state of the parent process; and

resuming execution of the child process such that an uninstrumented version of the child process is executed.

2 1
8. (Currently amended) The method of claim 7, wherein storing comprises retaining a copy of the uninstrumented instructions ~~replaced by the breakpoint~~ in a process image store.

3 1
9. (Currently amended) The method of claim 7, wherein monitoring comprises ~~executing a~~ the process monitor ~~configured to respond~~ responding to trace events generated by the child process.

10-19. (Canceled)

4 1
20. (New) The method of claim 7, wherein replacing comprises backpatching each occurrence of a breakpoint encountered in the address space of the child process.

5 1
21. (New) The method of claim 7, further comprising, after the child process has terminated, the process monitor setting a process monitor thread to monitor execution of the parent process.

6 5
22. (New) The method of claim 21, further comprising the process monitor:

monitoring the parent process for an indication that the parent process is about to resume execution;

suspending execution of the parent process in response to the indication that the parent process is about to resume; and

restoring each breakpoint occurrence to the address space of the parent process.

7 6
23. (New) The method of claim ~~22~~, further comprising the process monitor:

in association with the replacing, generating and storing a list of breakpointed functions such that a record is maintained as to what instrumented functions were replaced in the child process, wherein the restoring is performed by the process monitor in reference to the list.

8 6
24. (New) The method of claim ~~22~~, further comprising the process monitor:

resuming execution of the parent process; and

monitoring execution of the parent process.

9
25. (New) A computer-readable medium that stores a software tool, the software tool comprising:

an instrumentation engine configured to instrument a parent process to generate an instrumented version of the parent process that contains instrumented functions, the instrumentation engine further configured to store uninstrumented instructions associated with the instrumented functions in a process image store; and

a process monitor configured to monitor execution of the parent process and a child process created by the parent process, the process monitor further being configured to suspend execution of the child process upon an initial breakpoint being encountered, to replace each occurrence of a breakpoint in the address space of the child process with the uninstrumented

instructions from the process image store, and to resume execution of the child process such that an uninstrumented version of the child process will be executed.

10 9
26. (New) The computer-readable medium of claim 25, wherein the process monitor is further configured to monitor the parent process for an indication that the parent process is about to resume execution, to suspend execution of the parent process in response to the indication that the parent process is about to resume, and to restore each breakpoint occurrence to the address space of the parent process such that the instrumented version of the parent process will be executed.

11 10
27. (New) The computer-readable medium of claim 26, wherein the process monitor is further configured to generate and store a list of breakpointed functions such that a record is maintained as to what instrumented functions were replaced in the child process, wherein the process monitor restores each breakpoint occurrence through reference to the list.

12 10
28. (New) The computer-readable medium of claim 26, wherein the process monitor is further configured to resume execution of the parent process and monitor execution of the parent process.